Workshop **OMiLAB4FoF**

Interoperability through a model driven approach and application to Product Service Systems: MDSEA Architecture

June 27th 2016

Yves DUCQ
Professor
President of PGSO of IVlab

www.u-bordeaux.fr

Plan de la presentation

- ✓ MSEE Project - FoF-ICT-2011.7.3: Virtual Factories and enterprises
- ✓ Problem statement: Servitization principles and extended product
- ✓ How to ensure interoperability
- ✓ MDSEA architecture
  - ➢ BSM
  - ➢ TIM
  - ➢ TSM
- ✓ The SLM TOOLBOX and application case study
- ✓ Conclusions

MSEE Project

Project No:	284860
Project Full Name:	Manufacturing Service Ecosystem
Duration:	36 months
Start date:	October 1st 2011
Partnership:	19 partners, 9 countries
Strategic Objective:	FP7 FoF-ICT-2011.7.3 Virtual Factories and Enterprises
Total Eligible Cost:	15. 200.000 EURO
EC Contribution:	9.870.000 EURO

MSEE Project

VISION: By 2015, novel service-oriented management methodologies and the Future Internet universal business infrastructure will enable European virtual factories and enterprises to self-organize in distributed, autonomous, interoperable, non-hierarchical innovation ecosystems of tangible and intangible manufacturing assets, to be virtually described, on-the-fly composed and dynamically delivered as a Service, end-to-end along the globalised value chain.

COLLABORATIVE SERVICE INNOVATION

SERVICE ORIENTATION






BUSINESS COLLABORATION








MSEE Project

1. A **Service Lifecycle Management Toolbox** to model the various aspects of an enterprise during its servitization
2. A complete set of **Servitization Projects Templates** to be configured for any kind of Domain and Sector
3. An innovative **Servitization Maturity Model** for driving EU Manufacturing Enterprise towards advanced forms of servitization
4. A collaborative **Innovation Ecosystem Platform** to stimulate creativity and co-create service innovation through collaboration
5. An innovative set of **Enterprise Applications as a Service** to support the operation of advanced product-related services in an ecosystem
6. Four distinct but interlinked **Pilots and Experimentations** of the Manufacturing Service Ecosystem concept and IT infrastructure

Problem statement: servitisation

- ✓ European manufacturing enterprise will progressively migrate from traditional product-centric business to product-based service-oriented virtual enterprise and ecosystems
- ✓ The economy developed around the service related to concrete product is called PSS (Product Service System) or Servitisation
- ✓A lot of definitions and characterisation have been done for a service but few research works have been carried out concerning the service system

Problem statement: servitisation

- ✓IBM has characterised what is “service science”: *a growing multi-disciplinary research and academic effort that integrates aspects of established fields like computer science, operations research, engineering, management sciences, business strategy, social and cognitive sciences, and legal sciences*
- ✓ In the computer science domain, Service Oriented Architectures (SOA), have revolutionized information systems, by providing software engineers with powerful methodologies and tools for decomposing complex systems into autonomous components

Problem statement: servitisation

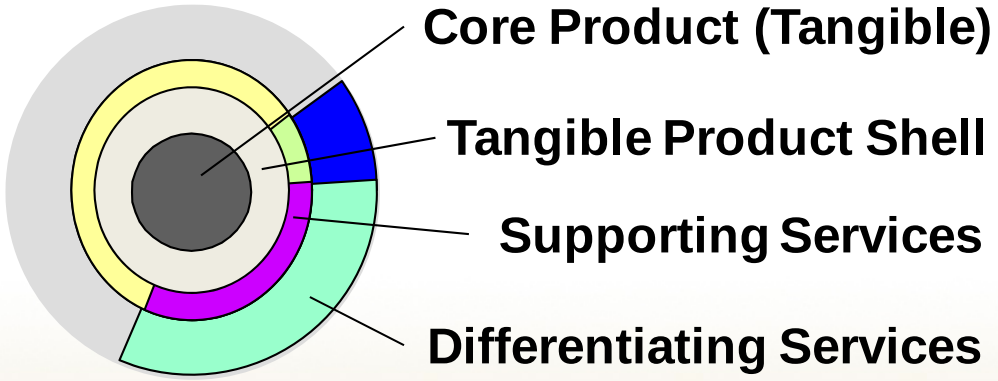
✓The servitization of manufacturing companies covers different levels of service provision and consequently different stages can be followed to evolve.

✓In servitization, the product is considered as the core element of the service to deliver to customers and subsequently we follow a manufacturing approach taking into account the market pressure that oblige to create new models in order to meet the servitization challenge

✓ An appropriate concept to link products, product related services and the needs of the users is the “Extended Product”

Problem statement: extended product

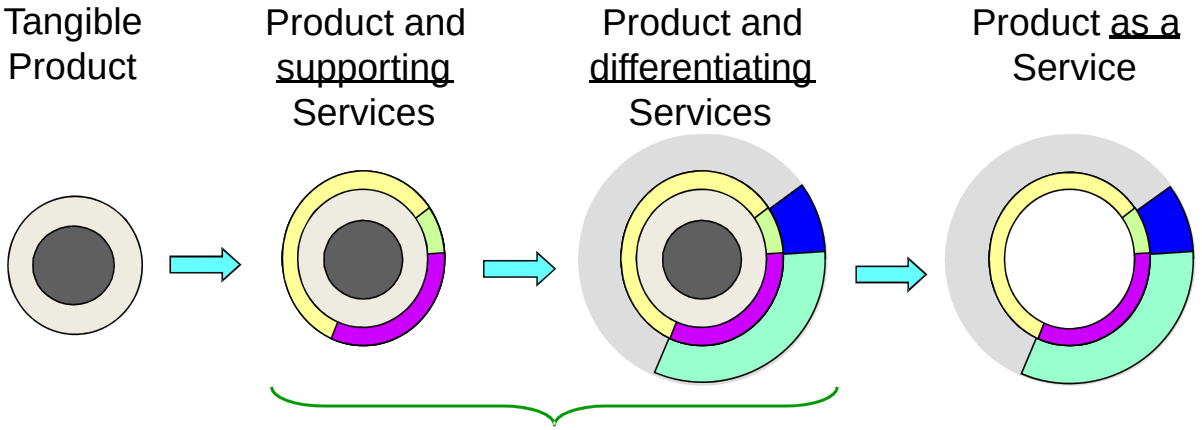
- ✓ The Extended Product concept belongs to the category of Product-Service System
- ✓ The Extended Product* is a complex result of tangible and intangible components



*Thoben, K.-D., Jagdev, H., Eschenbächer, J. (2001) Extended Products: evolving traditional product concepts.

Problem statement: EP and Servitisation

✓ How to migrate from traditional product to product as a service?



A machine tool

With maintenance

With specific services as performance measurement, optimisation

Product2 Service

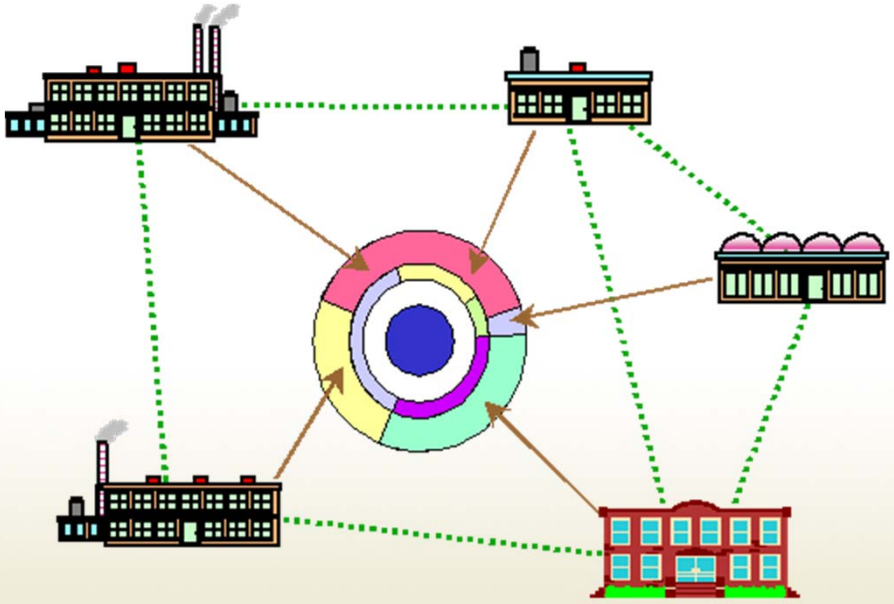
Hours of machine tool running with specific performances

How to make the service system evolve?

Problem statement: service system

✓ Enterprise cannot provide product and related services by its own

- need to collaborate
- need to build a virtual company



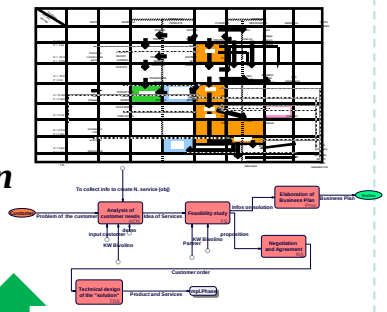
Problem statement: service system

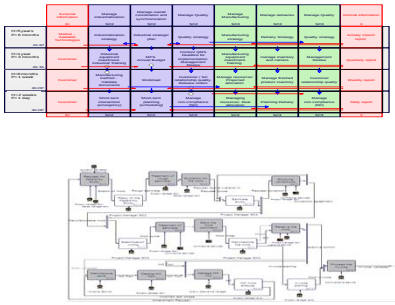
- ✓ But: the virtual enterprise can/should/must be different according to the different phases of the service life cycle
- ✓ Need to define the service life cycle phases
- ✓ Need to have a manufacturing and service ecosystem
  - ➢ **need to ensure the interoperability between VME's**

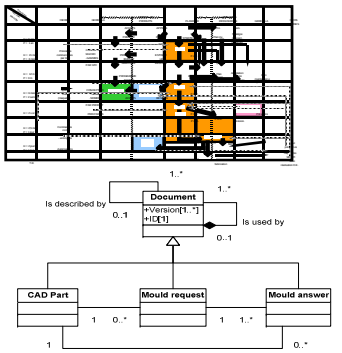
Service life cycle

Service requirement phase | Service design phase | Service Development phase | Service realisation phase

Virtual Organisation Equivalent models

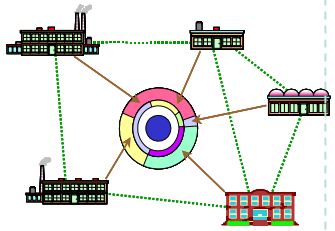




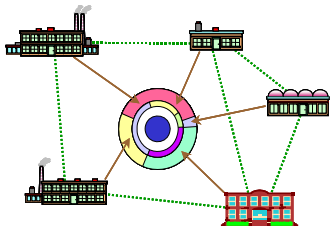


modelling

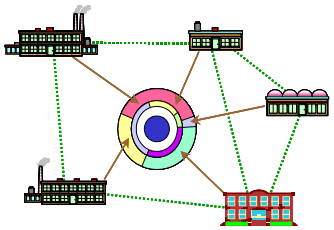
Real organisations



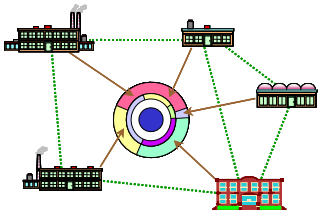
Virtual organisation for service requirements definition



Virtual organisation for service design



Virtual organisation for service development



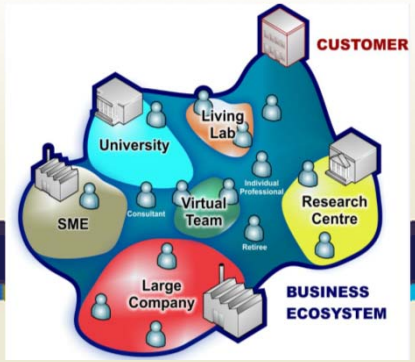
Virtual organisation for service realisation

Virtualisation

Virtualisation

Virtualisation

Virtualisation



First topic: Alignment

- How to ensure that the service system implemented is coherent with the enterprise strategy
  - In terms of objectives
  - In terms of technical performances
  - In terms of social performance and in particular psycho-social risk prevention

First topic: Alignment

- Necessity to have a performant method to collect **requirements** in a **participative** approach with users
- Model driven approach
  - Which kinds of modelling levels to ensure the continuous modelling from user to technical points of view
  - How to ensure model transformation between modelling level
  - How to go from modelling to IT-HR-PM development

Second topic: Interoperability

- How to ensure that the implemented system is interoperable with other partners
  - To ensure that IT and Practices are interoperable
  - to ensure sustainable and long term development (from economic and enviromental points of view)
  - but with unsustainable and short term collaborations

Second topic: Interoperability

- Adaptability of the system components
- How the take the human aspects more into account
- Role of ontology in the modelling of the various systems
- From an economy of owner to an economy of usage

How to ensure interoperability: Service system modelling

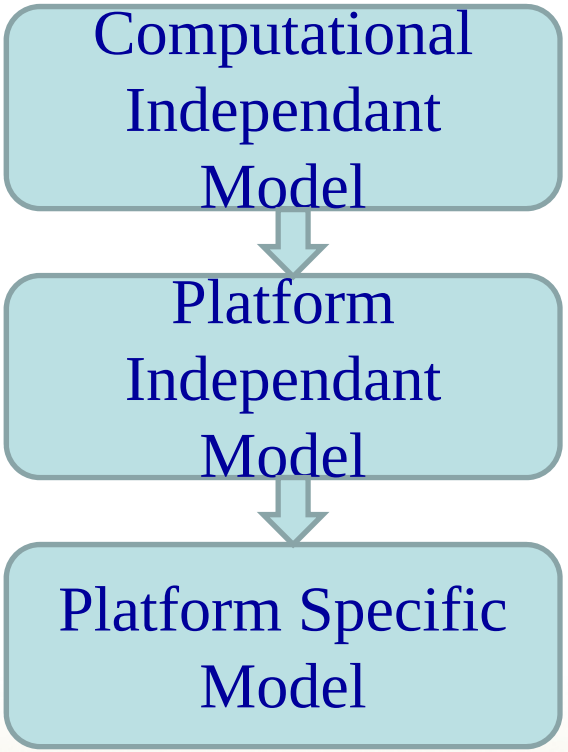
- ✓ Need to understand why and how the virtual enterprises will be organised
- ✓ Need to be sure that the organisation and components of the virtual enterprise will be coherent with the objectives of the VE
  - ➢ need to use enterprise modelling **but not to develop a new language**
  - ➢ need to have a progressive approach in the modelling from the business definition to the detailed implementation

Why a Model Driven Approach

- ✓ To start from the users points of view (business level)
- ✓ To separate and share the preoccupations from users to technique
- ✓ To have a set of coherent modelling levels which are based on system theory
- ✓ Finally to facilitate the alignment between the business view and the technical development
- ✓ At each level, to validate more and more detailed specifications until the implementation

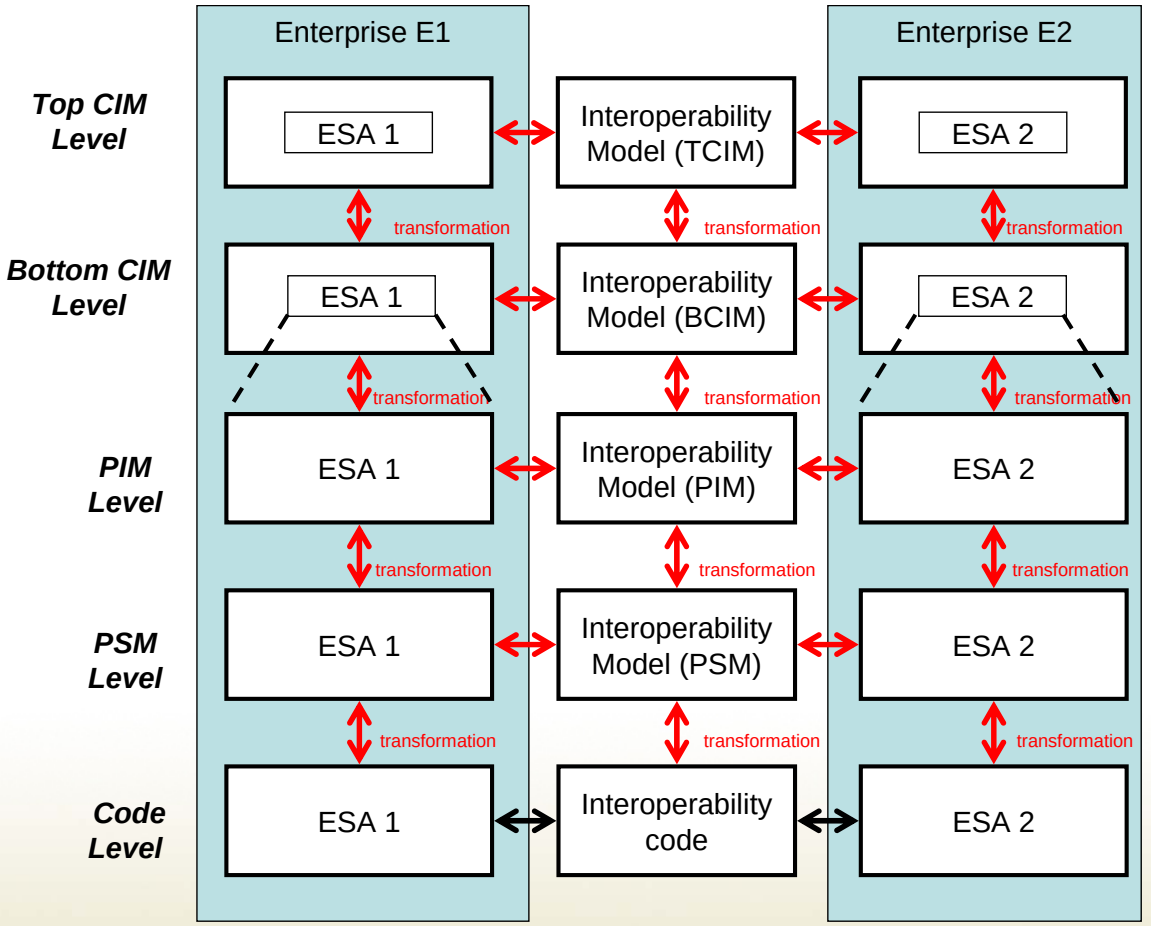
Background in Model Driven Approach

- ✓ Model Driven Architecture (MDA)
  - ➢ Developed by Object Management Group
  - ➢ First architecture in this sense
  - ➢ Explain what but not how



Background in Model Driven Approach

- ✓ Model Driven Interoperability (MDI)
  - ➢ Developed first in the frame of INTEROP NoE
  - ➢ Improved in French projects on interoperability: ASICOM and ISTA3
  - ➢ Oriented towards IT in conjunction with SOA

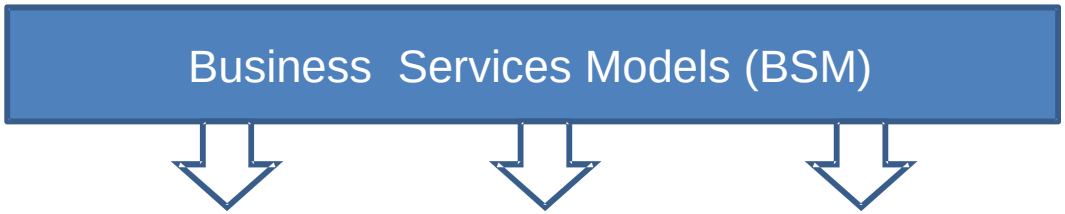


MDSEA

- ✓ The proposed Model Driven Service Engineering Architecture is elaborated based:
  - ➢ On MDA and MDI
  - ➢ On several modelling levels
- ✓ MDSEA must be adapted to services systems: to implement in VME IT, Organisation and Physical means
- ✓ MDSEA must define which kinds of modelling languages will be used
- ✓ MDSEA must define the transformation mechanisms

MDSEA

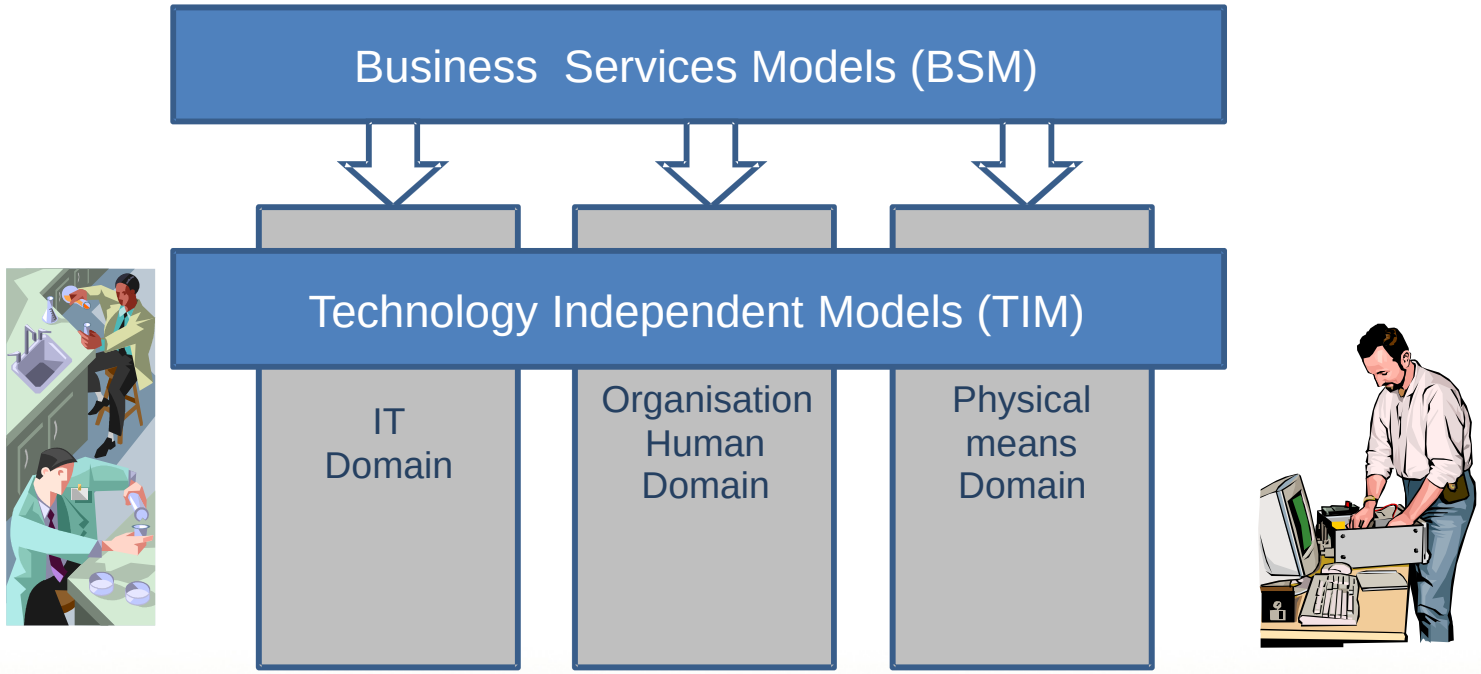
- **Architecture for Service System Engineering: Model Driven Service Engineering Architecture (MDSEA)**



- **Business Service Models (BSM): VME USER Oriented** -> models at the global level. The models at the BSM level must be independent to the future technologies that will be used for the various resources.
- Languages must be understandable and usable (to update the models) by users of the system
  - Languages must be simple but powerful
  - Languages must represent several points of view: service process and service process control for the design and the operation of the service system
- Chosen languages are Extended Actigram* and GRAI Grid and nets

MDSEA

- **Architecture for Service System Engineering: Model Driven Service Engineering Architecture (MDSEA)**

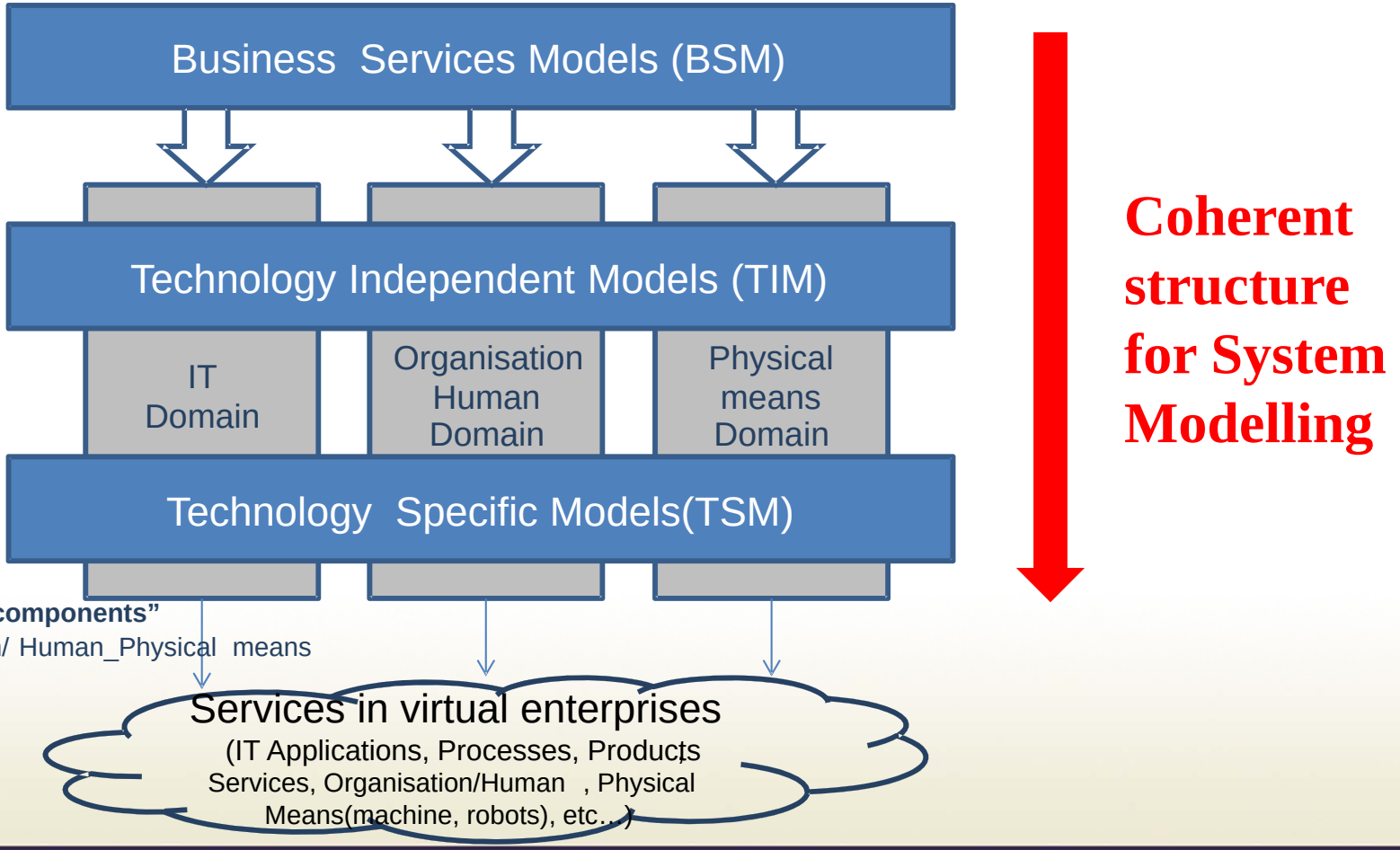


MDSEA

- **Technology Independent Models (TIM): First technical level :** The models at the TIM level must provide sufficient details to allow developing or buying software applications, components, recruiting human operators / managers or establishing internal training plans, buying and realizing machine devices, for supporting and delivering services in interaction with customers.
- Chosen languages are BPMN 2.0 and UML class diagrams

MDSEA

- **Architecture for Service System Engineering: Model Driven Service Engineering Architecture (MDSEA)**

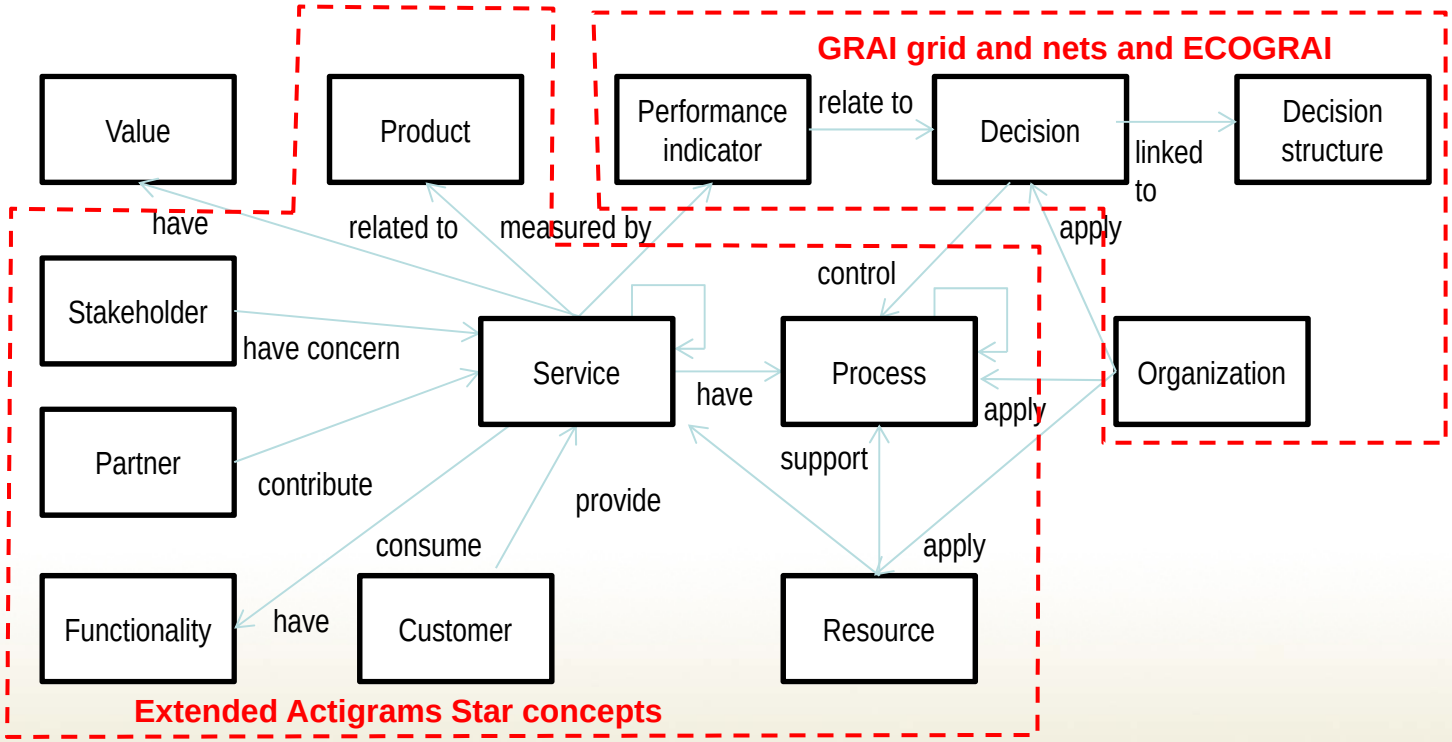


Business Service Modelling level

- ✓ Business Service is the first level of modelling
- ✓ Global description of the virtual enterprise
- ✓ Independent to the future technologies of resources
- ✓ Can be decomposed into 2 sub levels
  - ➢ top BSM: global
  - ➢ bottom BSM: domain concerns by the servitisation

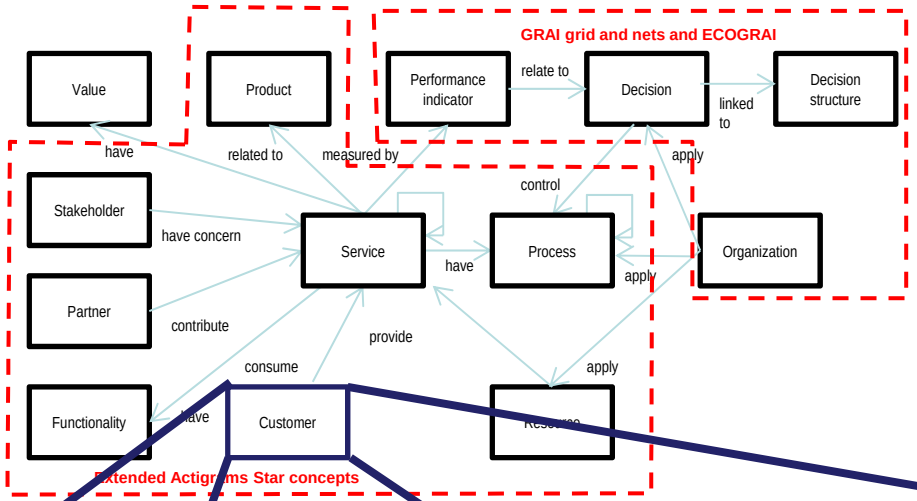
Business Service Modelling level

✓ Specific concepts must be identified and represented at the BSM level:



Business Service Modelling level

✓ Then, each concept can be detailed using a template:



Header	
Construct label	[‘Customer]
Identifier	[Identifier of the customer instance]
Name	[name of the customer instance]
Body	
Categories	[Worker, farmer, student,…]
Level of education	[short textual description]
Gender and age	[Short textual description]
Annual revenue range	[Short textual description]
Targeted frequency of service use	Short textual description]
Wish and requirement	[Short textual description]
Constraint	[Short textual description]
Relationships to other model elements	
SERVICE	[Identifier/name of Service concerned by the customer: described by Service template]
Other Relationships	
RELATED TO MODEL LEVEL	[Refer to BSM, TIM, TSM modeling level] : BSM
RELATED TO SLM PHASE	[Refer to service lifecycle phases] : Requirement

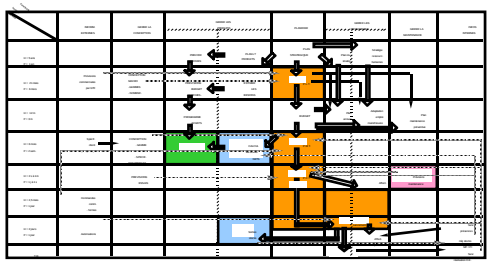
MDSEA

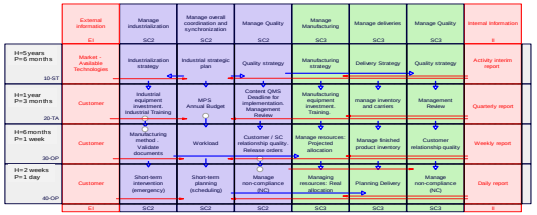
Examples of models at BSM level

Collaborative decisions

Collaborative decisions

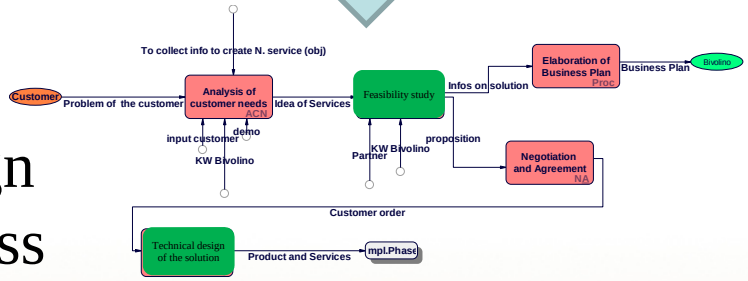
Design control

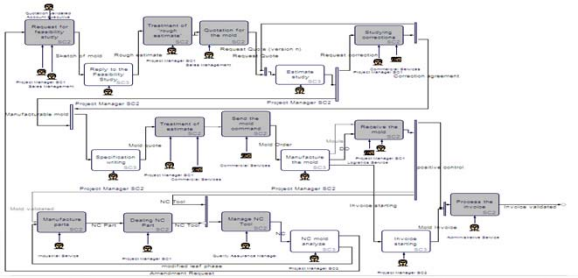




Operation control

Design process





Operation process

Collaborative process

Collaborative process

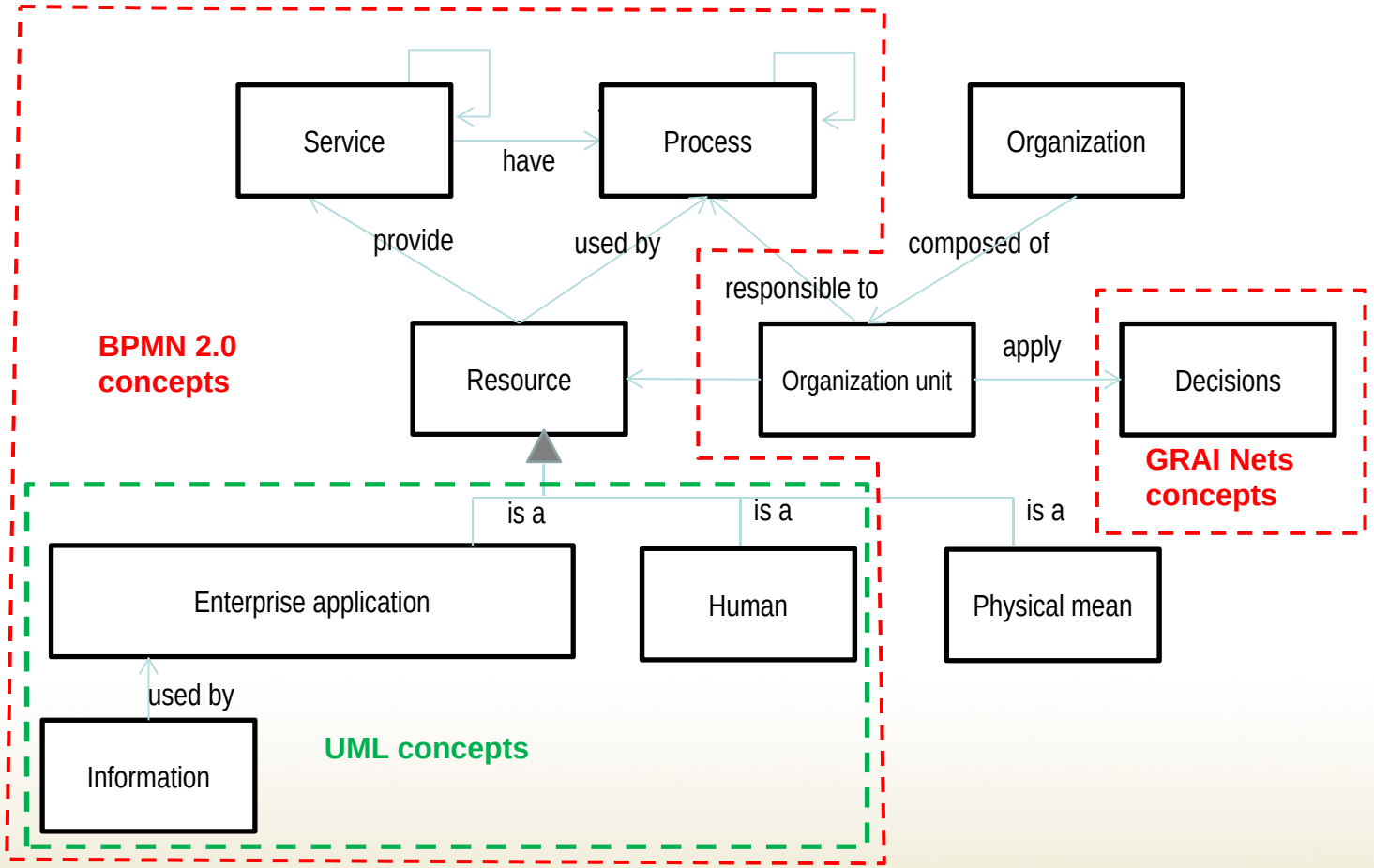
Technology Independent Modelling level

- ✓ Second level of abstraction in the representation of the service system
- ✓ Gives detailed specifications of the structure and functionalities of the service system but not propose technological details
- ✓ the resources specifications are described for
  - ➢ IT
  - ➢ organisation/human
  - ➢ physical means

Technology Independent Modelling level

- ✓ The functionalities are derived from BSM models
- ✓ Complementary data can be useful to collect
- ✓ the functionalities can be classified by importance in order to serve the selection of resources at the lower level
- ✓ the functionalities are also covering the interoperability problems to be connected to other companies of the virtual enterprise or to other companies of the ecosystem

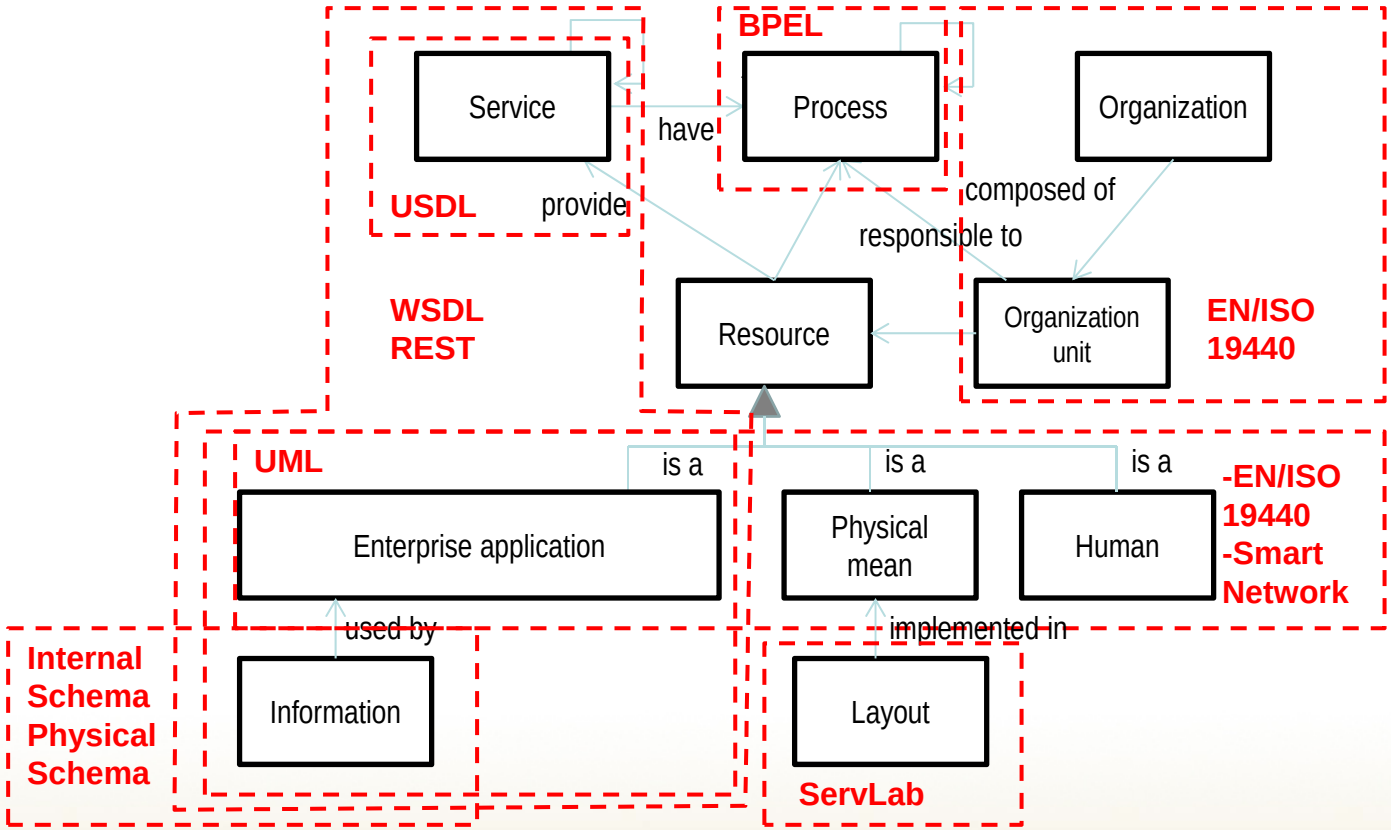
Technology Independent Modelling level



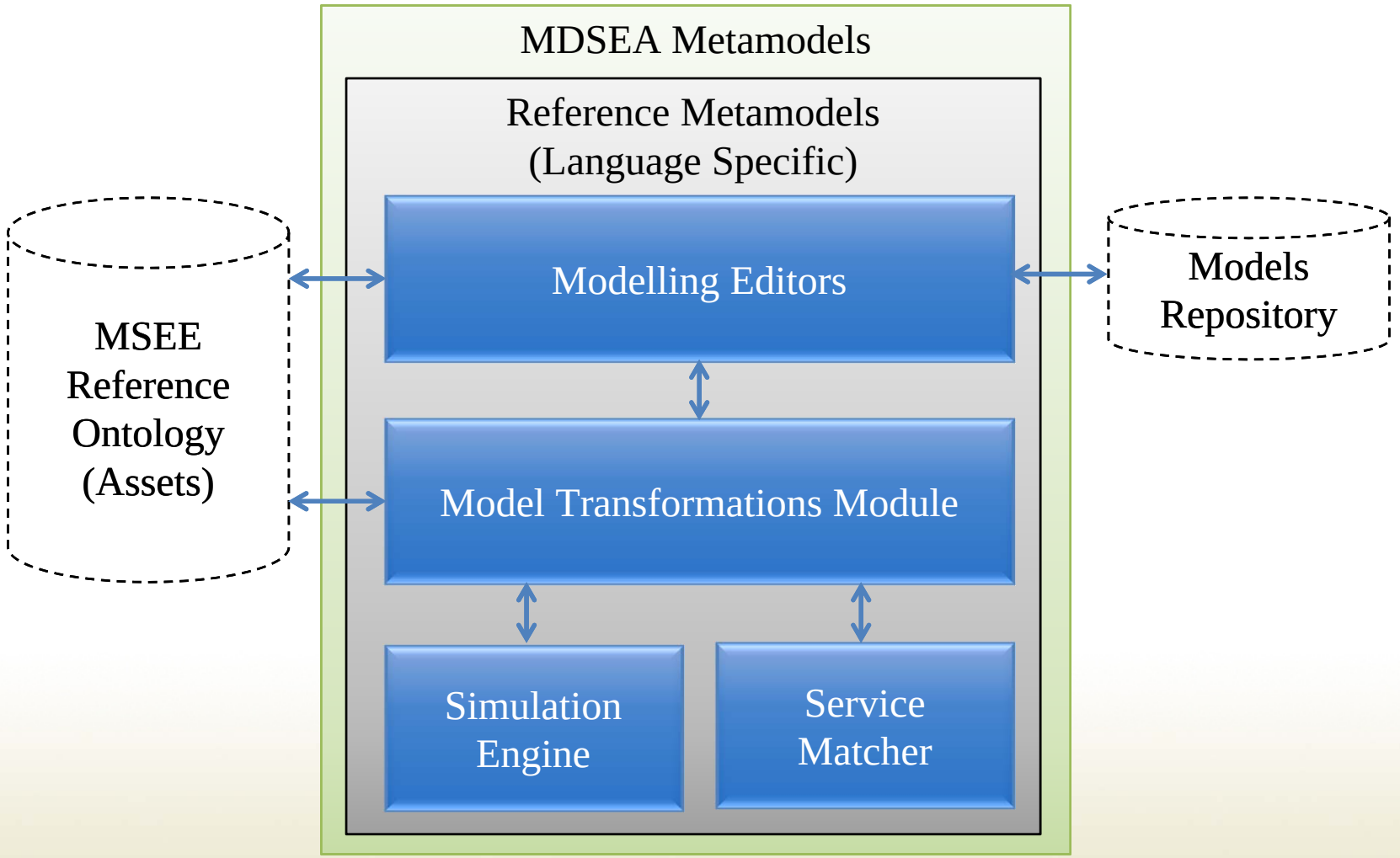
Technology Specific Modelling level

- ✓ Details how the system will use each resource
  - ➢ To choose how to buy or to develop and IT solution
  - ➢ To choose a human resources and to define him/her role and place in the organisation
  - ➢ To select a specific machine and to describe its performance

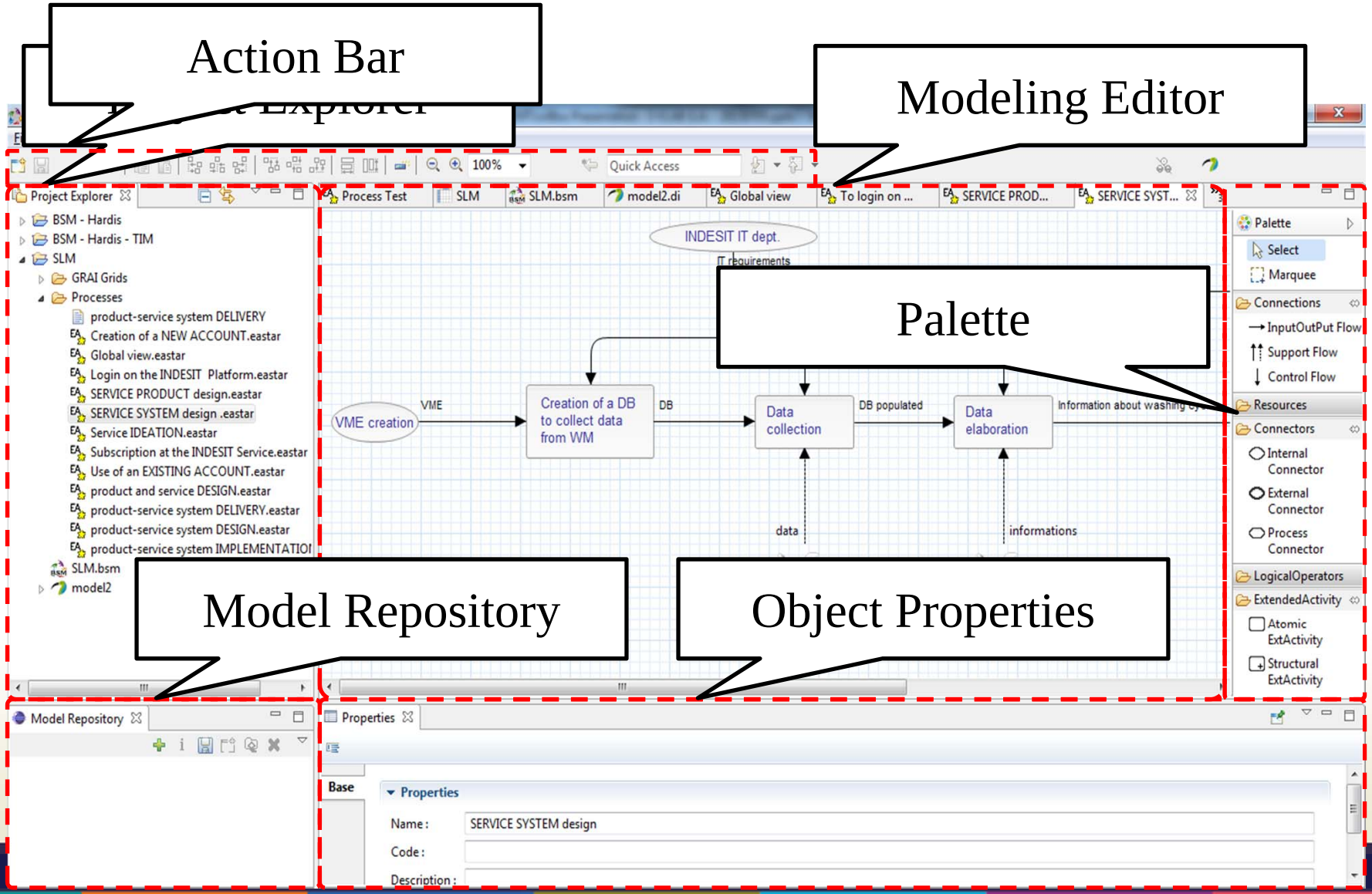
Technology Specific Modelling level



Platform for service and system modelling



Modelling editor: SLM TOOLBOX



Modelling languages at TIM Level

TIM Modelling iteration plan

Based on models at BSM

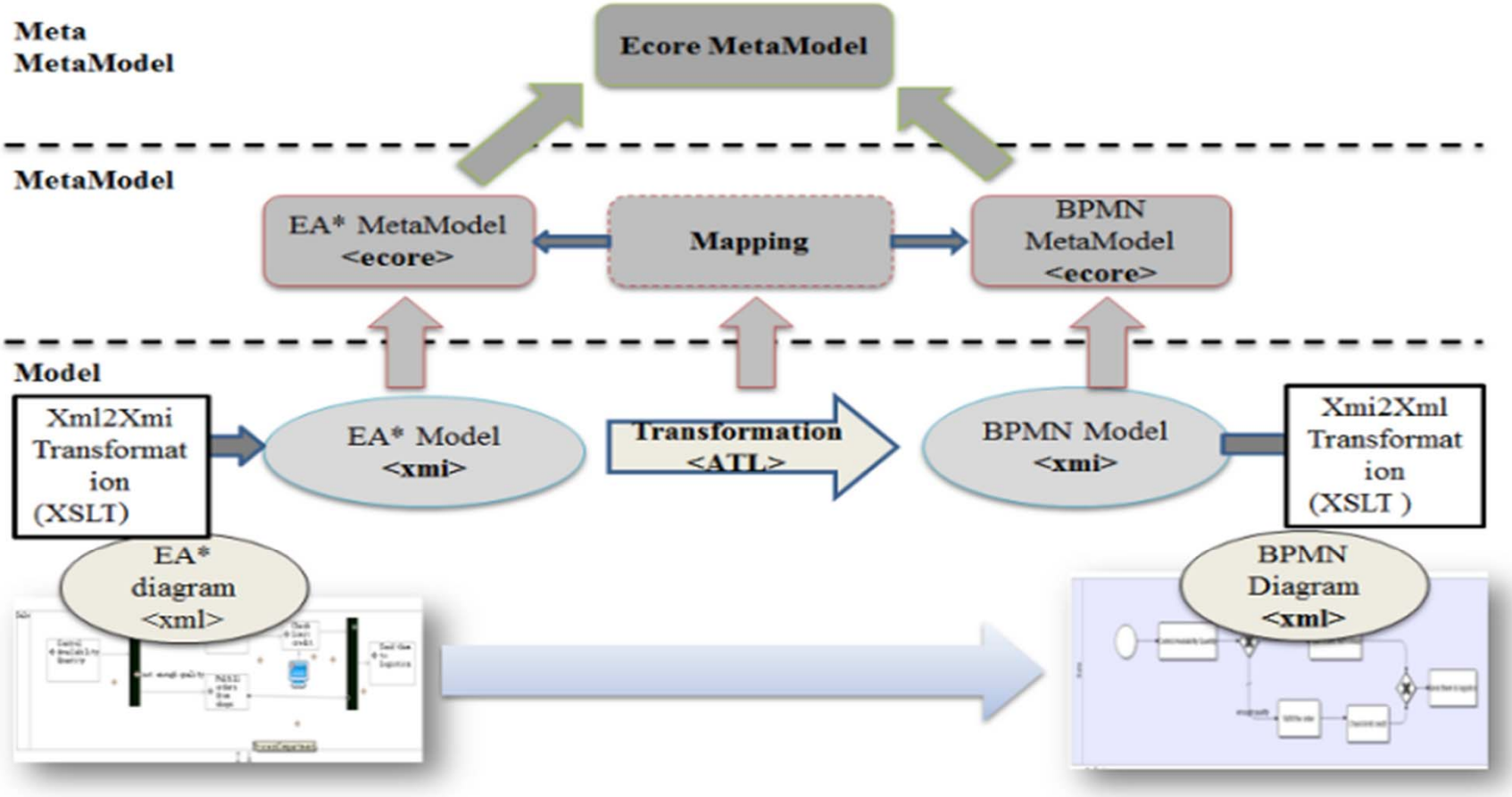
	Interaction View	Architectural View	Behavioral View	Structural View
Iteration #1 ***Goal : to validate the requirements***	• Use cases (UML Use Cases Diagrams) User centered processes (BPMN)	• General architecture diagram		
Iteration #2 ***Goal : to specify the functionalities***	• User Interface flow diagrams	• Domain Model (UML Class Diagrams)	• System centered processes (BPMN)	
Iteration #3 ***Goal : to design the architecture of the system***	• User Interface graphical design	• Logical architecture (UML Component Diagrams)		• Data model (UML Class Diagrams)
Iteration #4 ***Goal : to prepare the software design***			• Processings (UML Sequence Diagram)	• Object Models (UML Class Diagrams)

⇒ At any iteration, models can be annotated with terms from the Ontology

⇒ At any iteration, models can be stored in the modelling repository

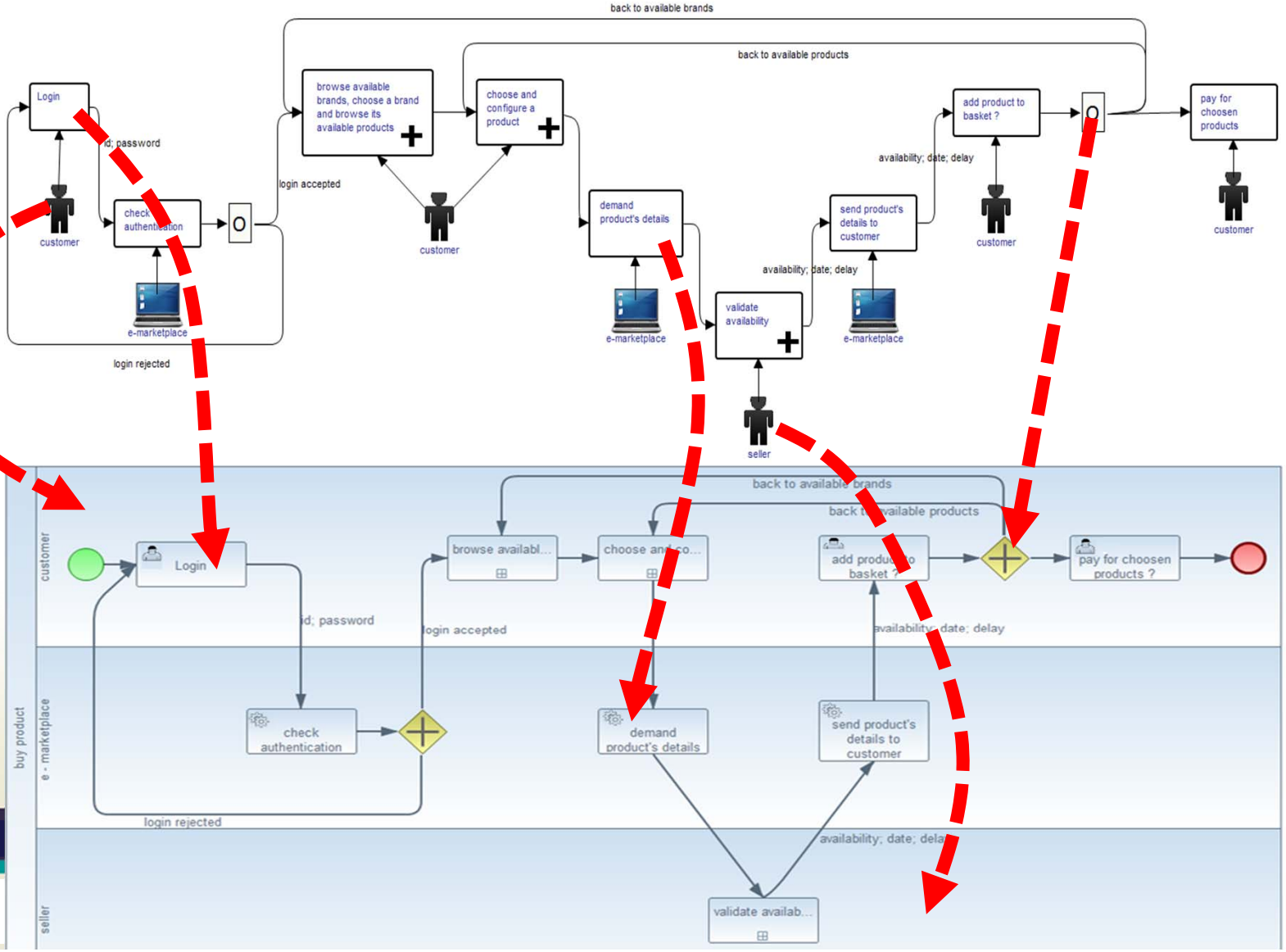
Model transformation principles

- **Different languages at different level of modelling**
  - need to transform one model in another model without too many supplementary information from users



Example of model transformation

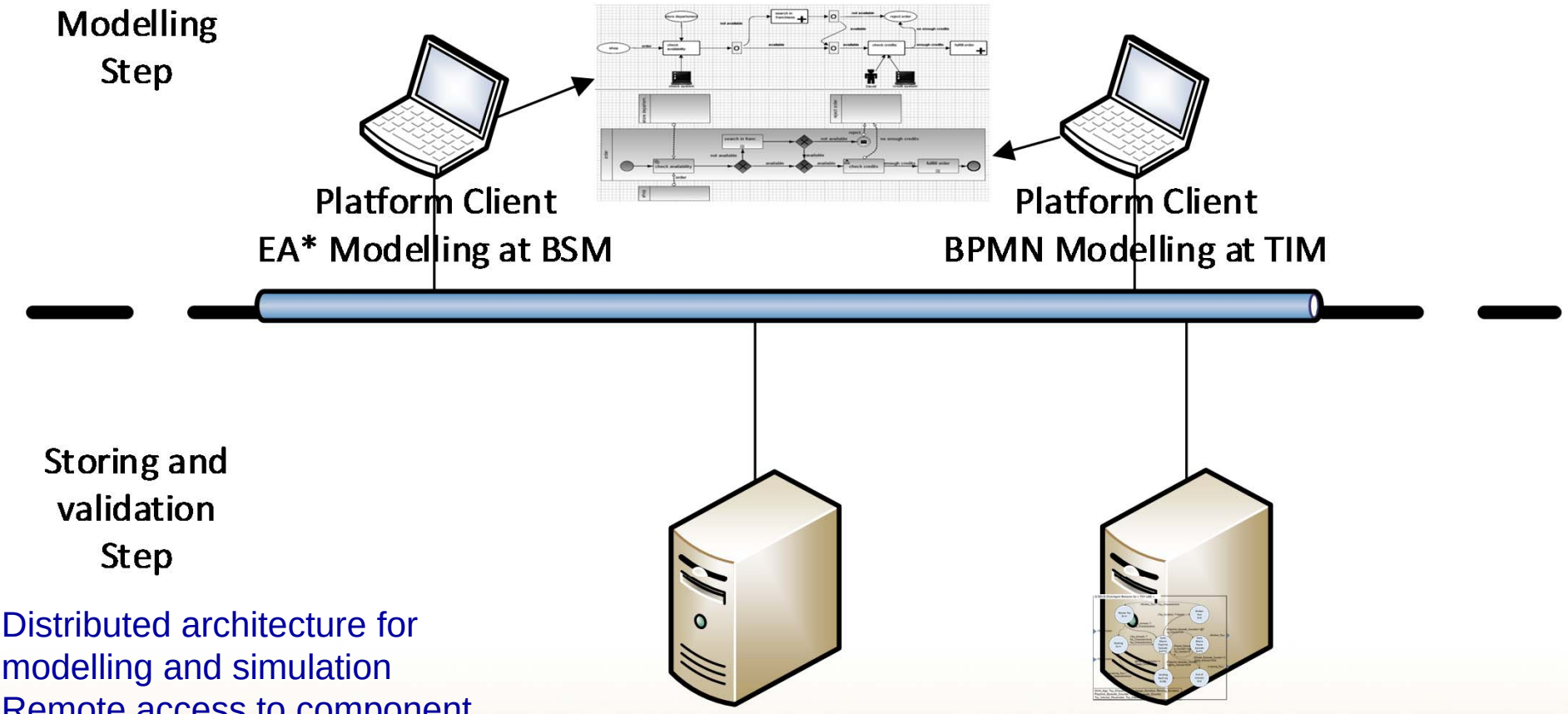
From EA* to BPMN



EA model for BIVOLINO at BSM*

BPMN 2.0 model for BIVOLINO at TIM

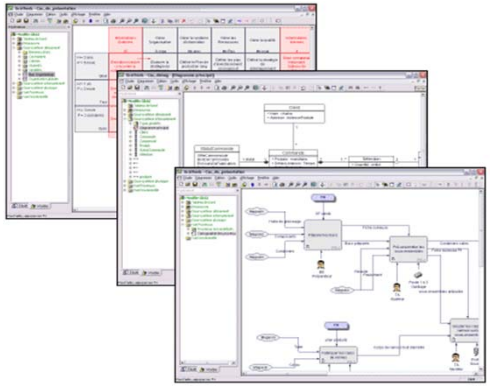
From BPMN to simulation



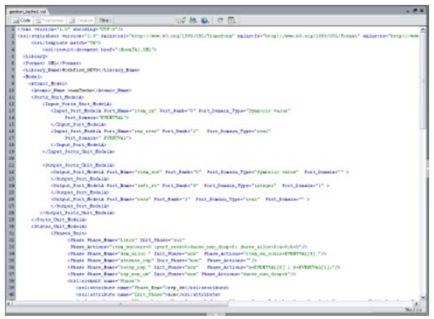
Distributed architecture for modelling and simulation
Remote access to component library
REST is used for access to the simulation model data base

From BPMN to simulation

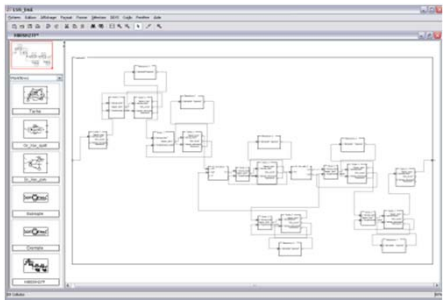
Extended Actigram Star



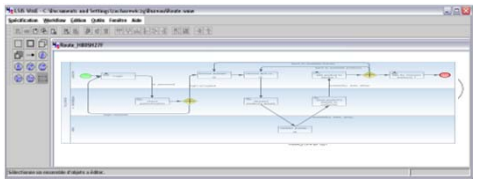
ATL Model Transformation



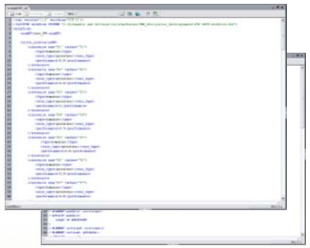
LSIS_DME Model Editor & Simulator



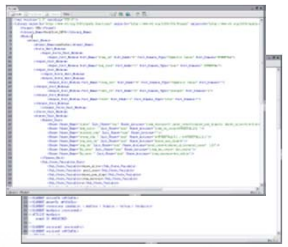
BPMN2.0

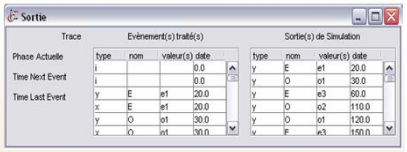


BPMN2.0 XMI



DEVS XMI

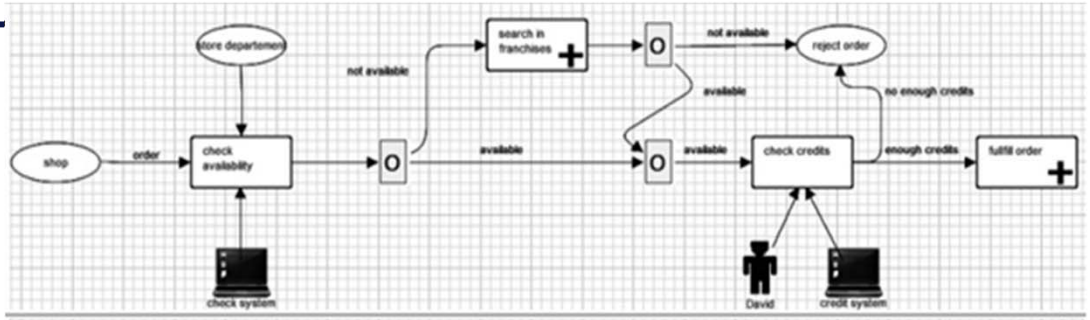


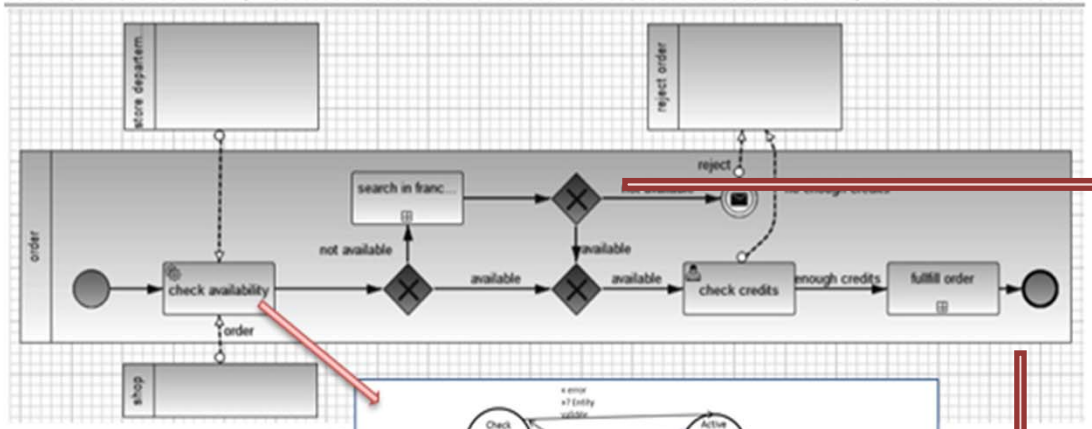


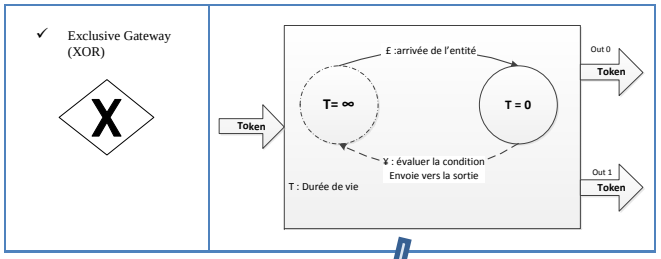
Use of results to display and animate BPMN model for non expert

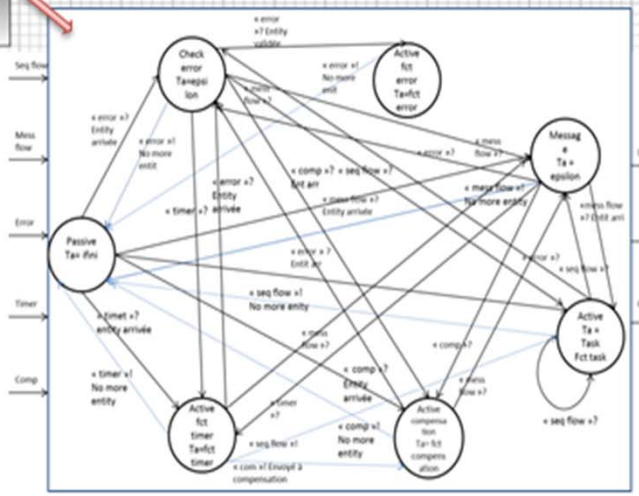
Simulation results

From BPMN to simulation

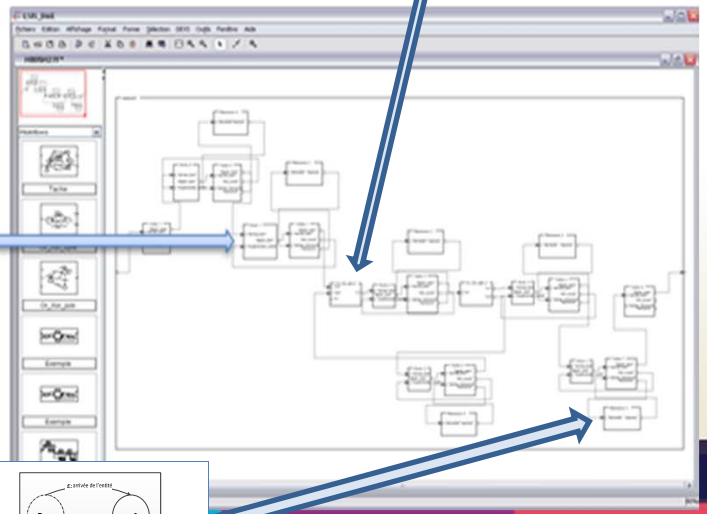


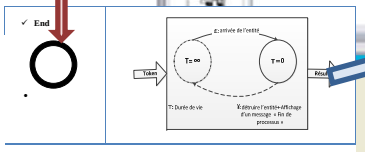






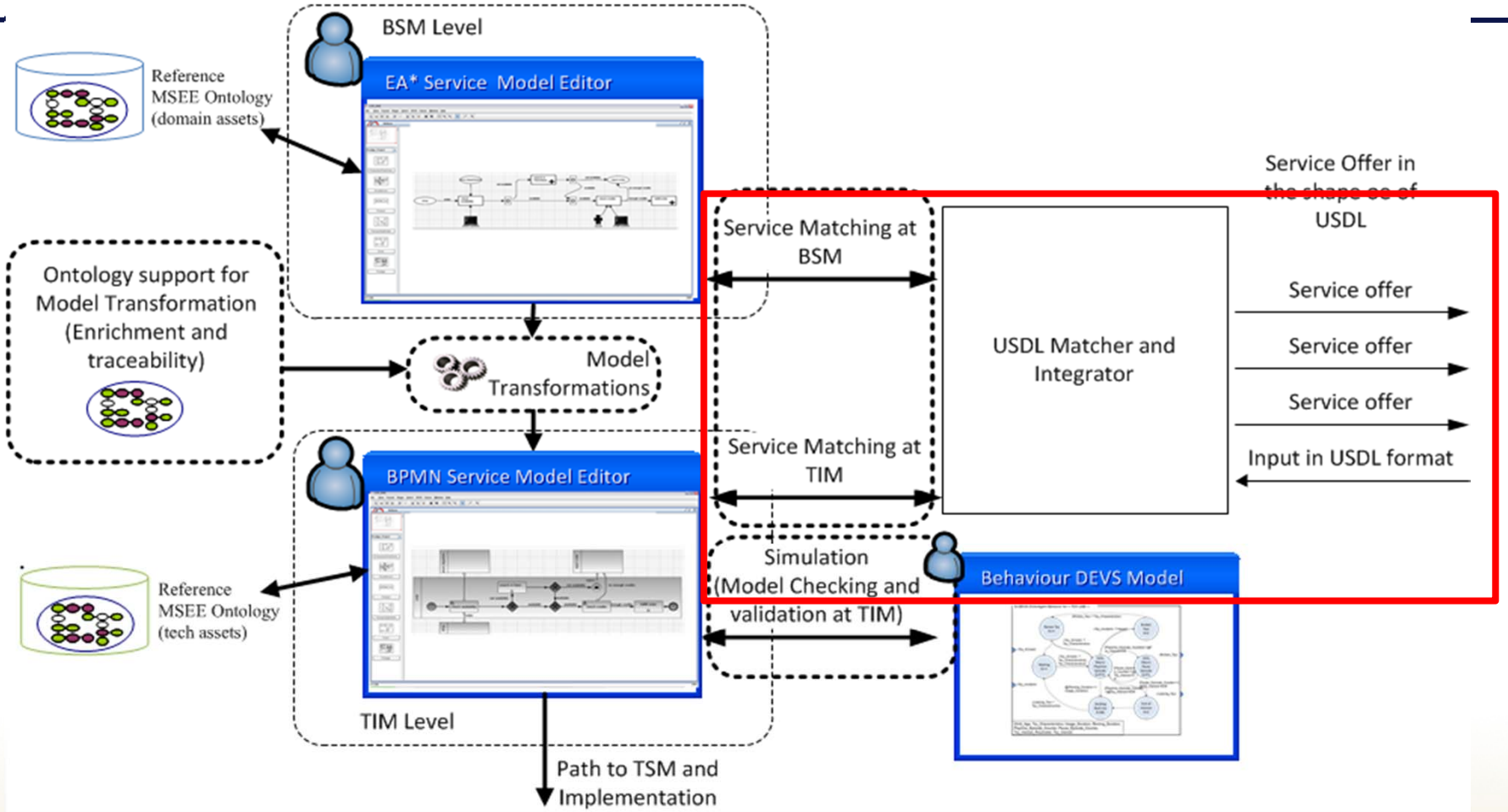
DEVS atomic model





From each component of BPMN standard a DEVS equivalent model has been created

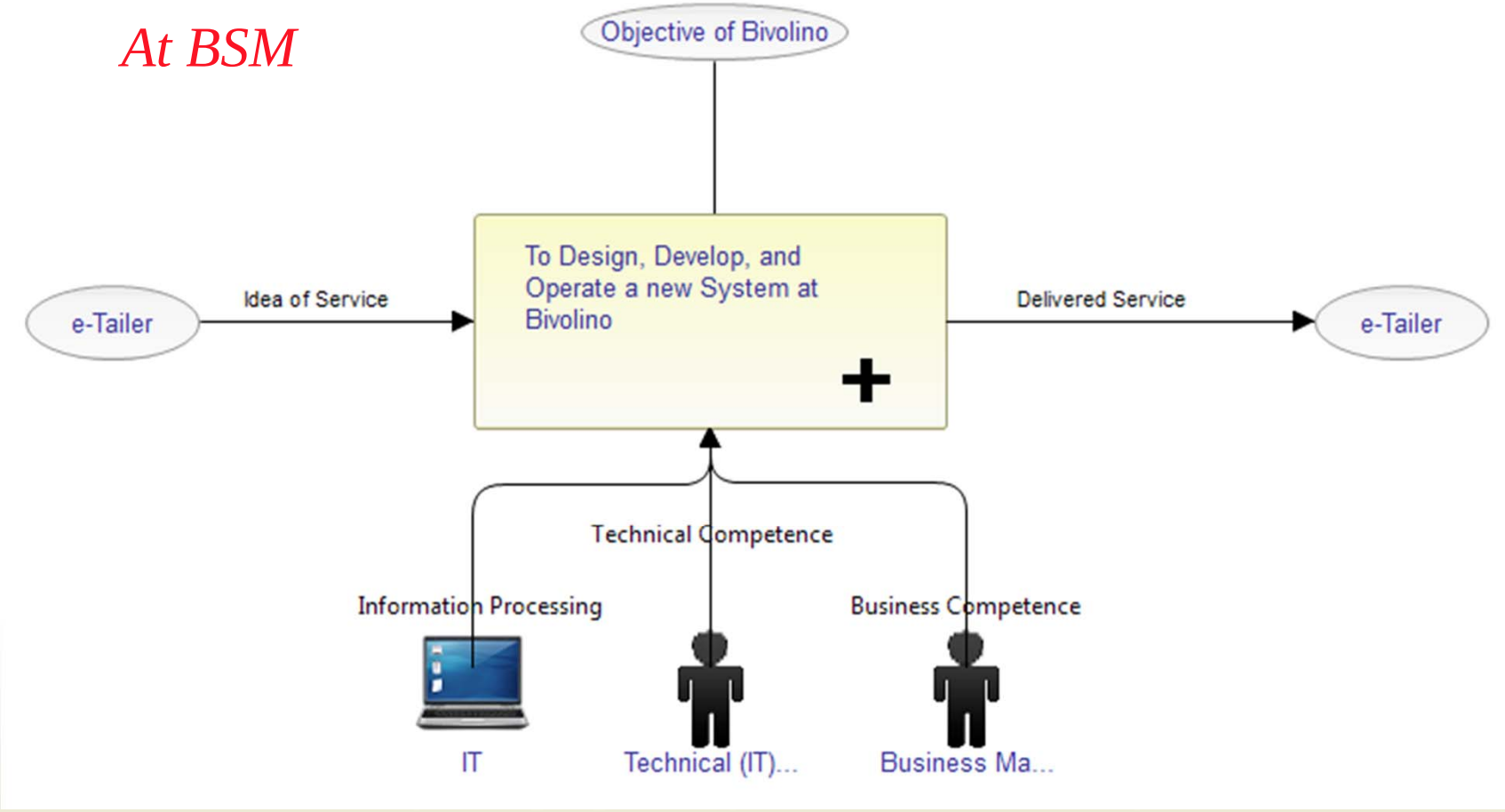
Service matcher with USDL



Ontology ensures Model enrichment and tractability during modelling and transformations
Simulation ensures model checking and performance indicators following regarding time
USDL ensures interoperability with other modelling platform in the domain

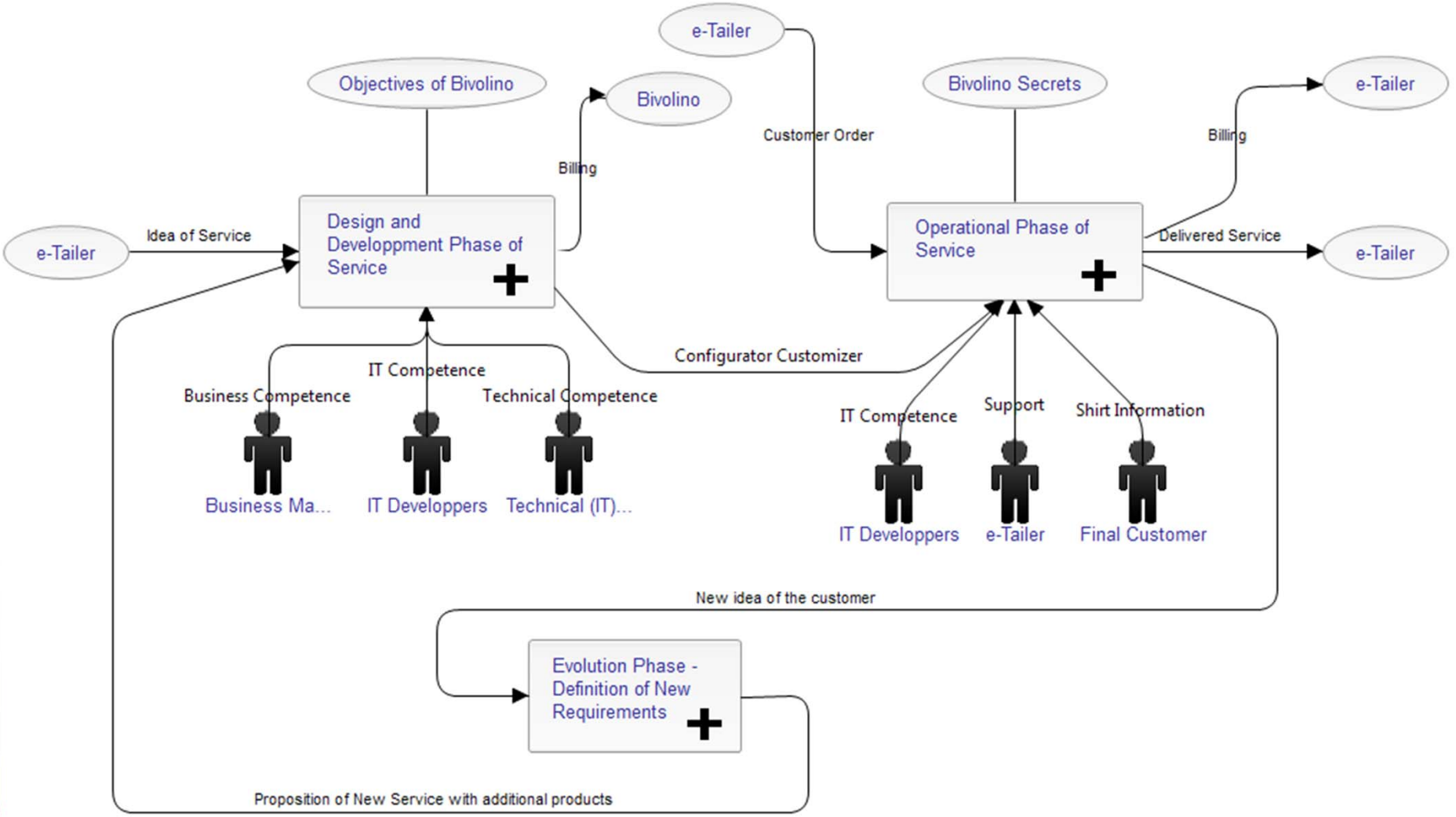
Example of BIVOLINO

At BSM



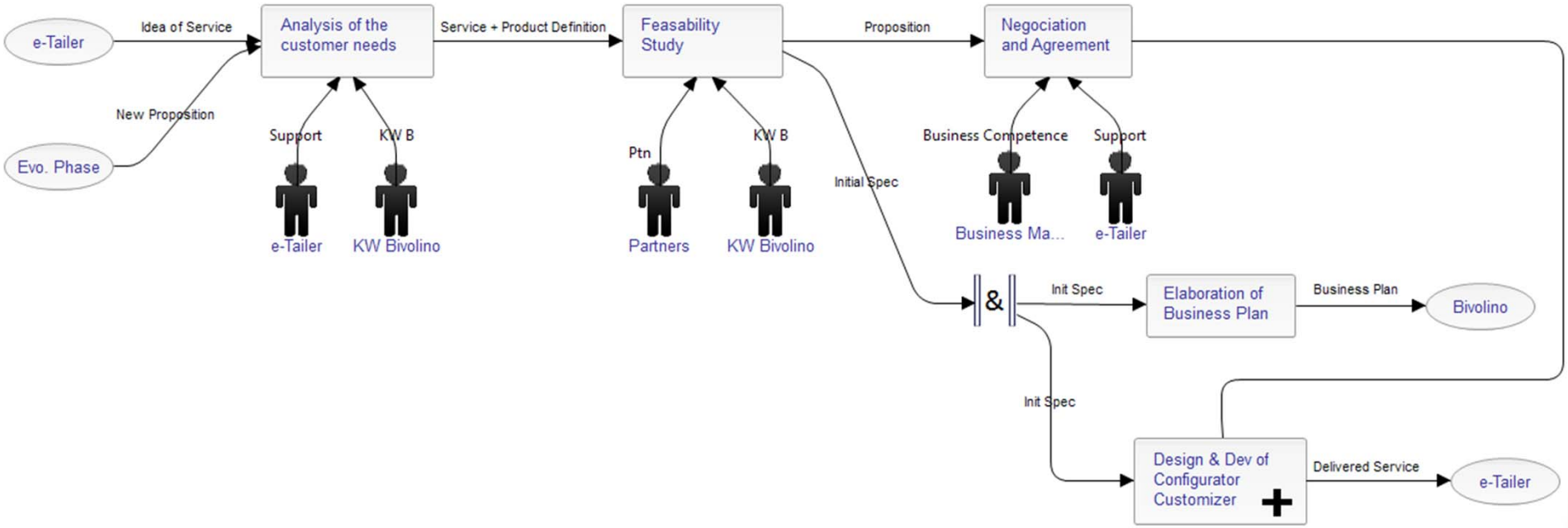
Example of BIVOLINO

To design Develop and Operate



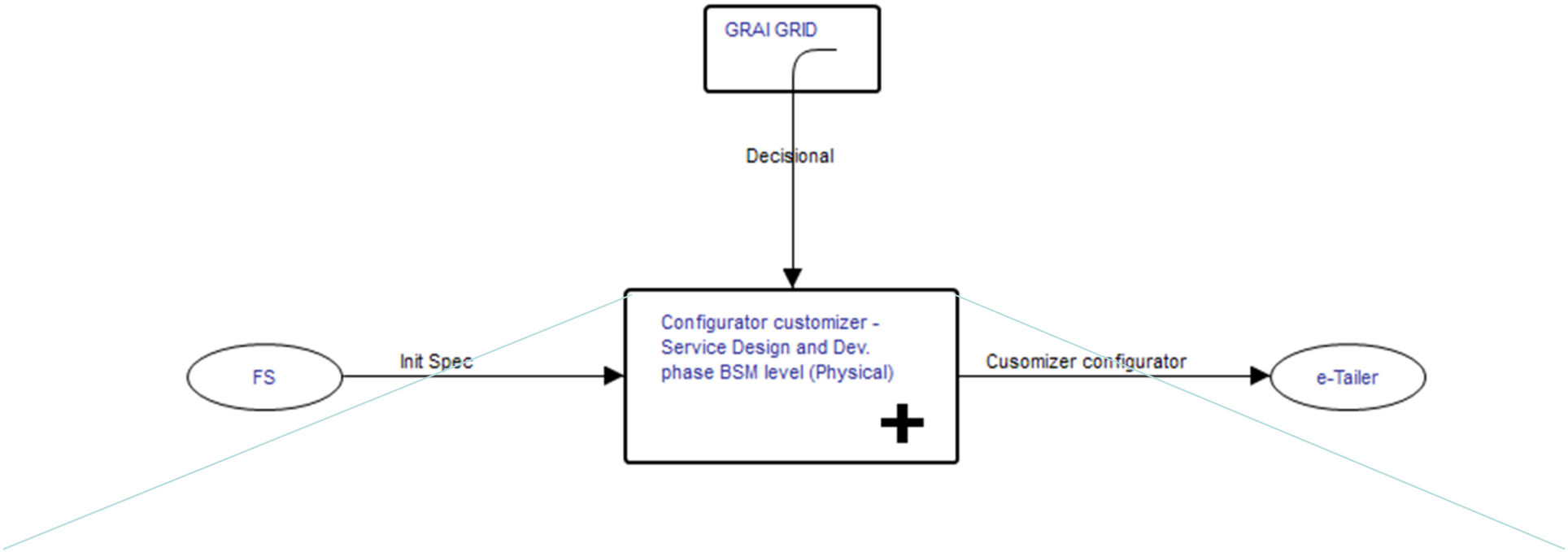
Example of BIVOLINO

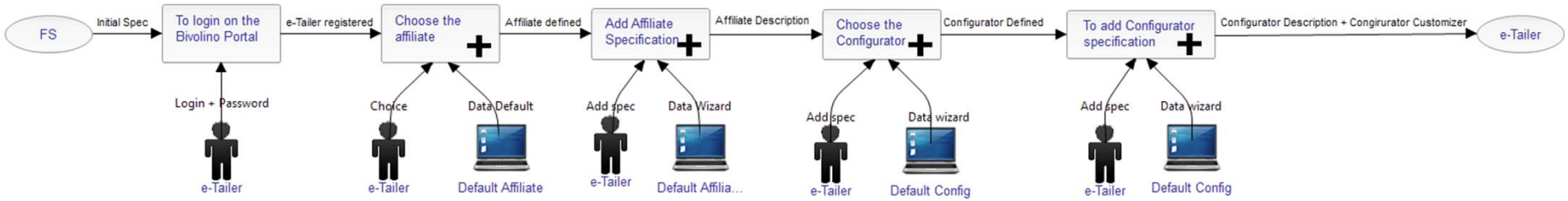
Design & Dev. Phase



Example of BIVOLINO

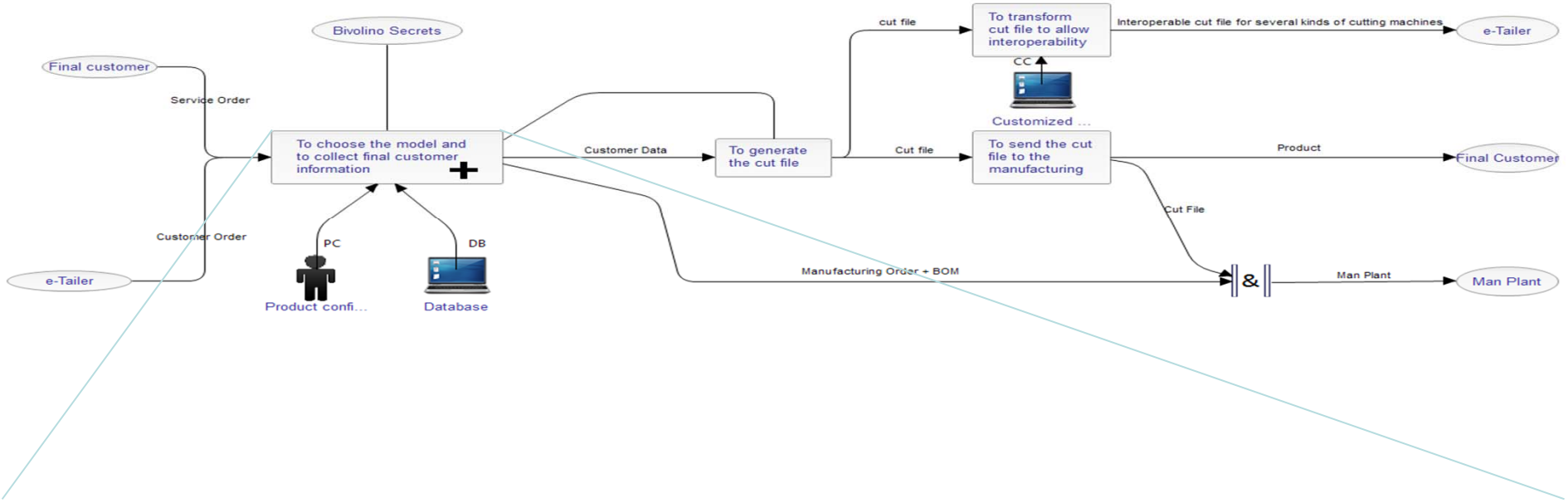
Design & Dev of Configurator Customizer

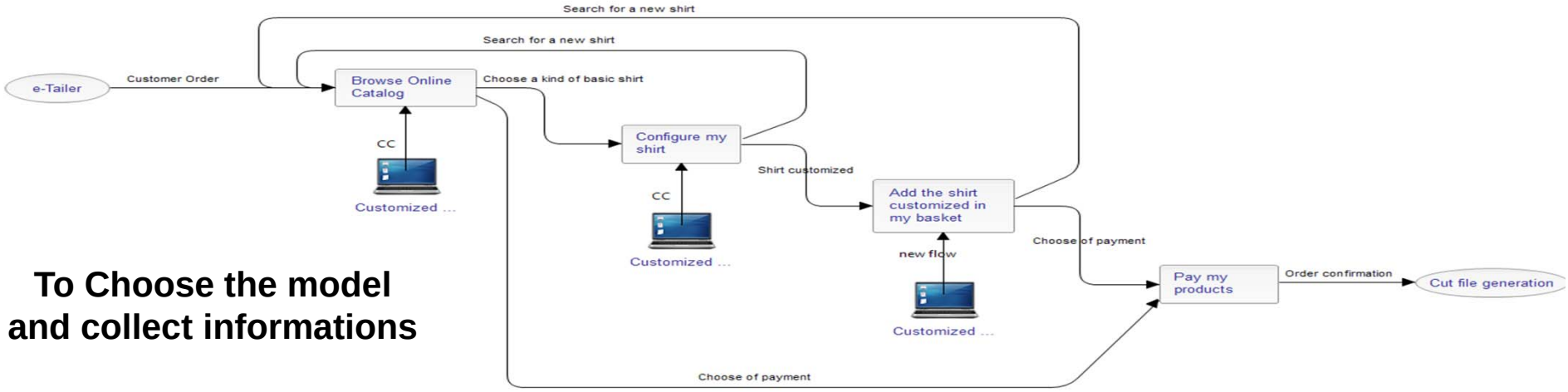




Example of BIVOLINO

Service Operational phase BSM level (Physical)

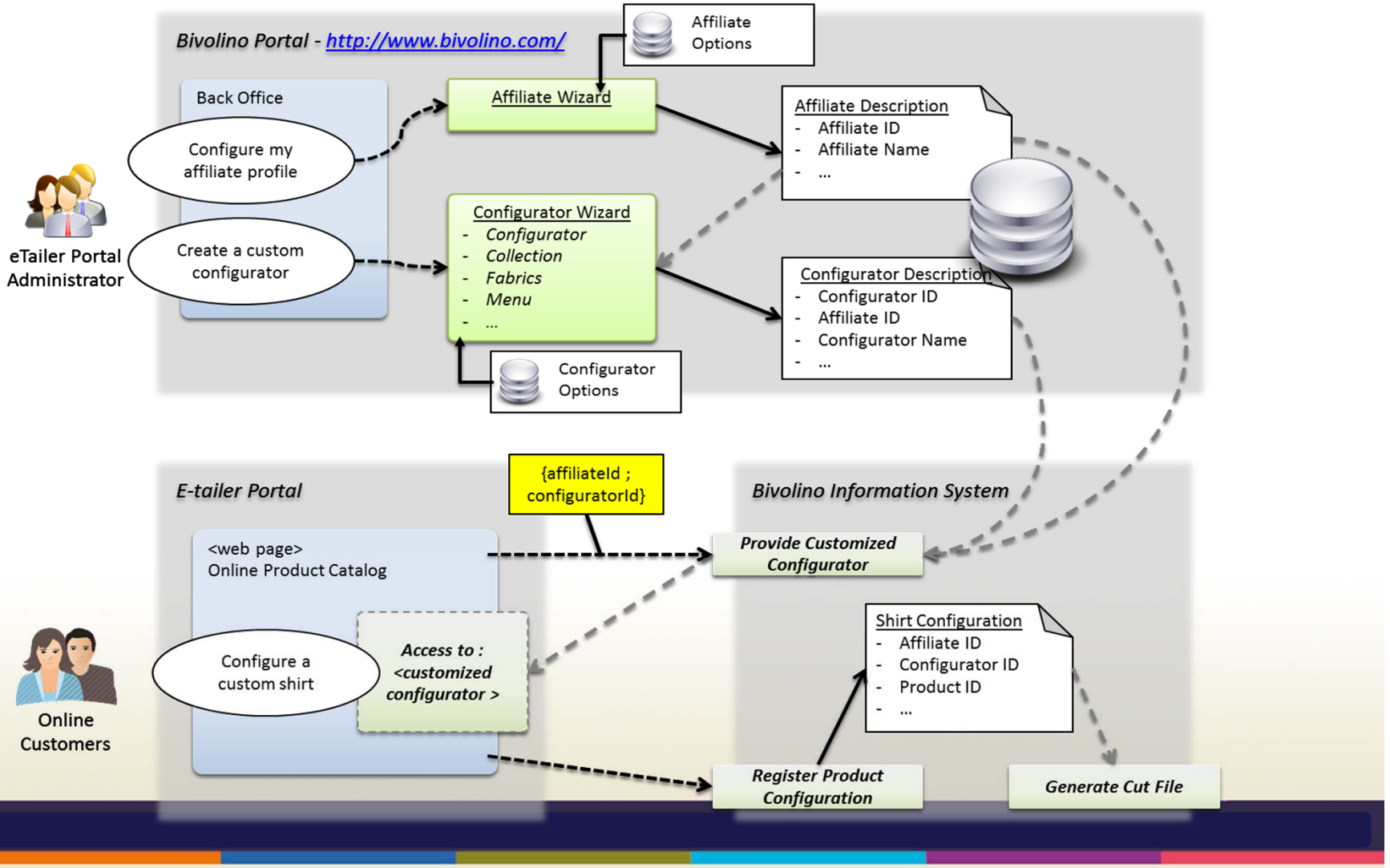




To Choose the model and collect informations

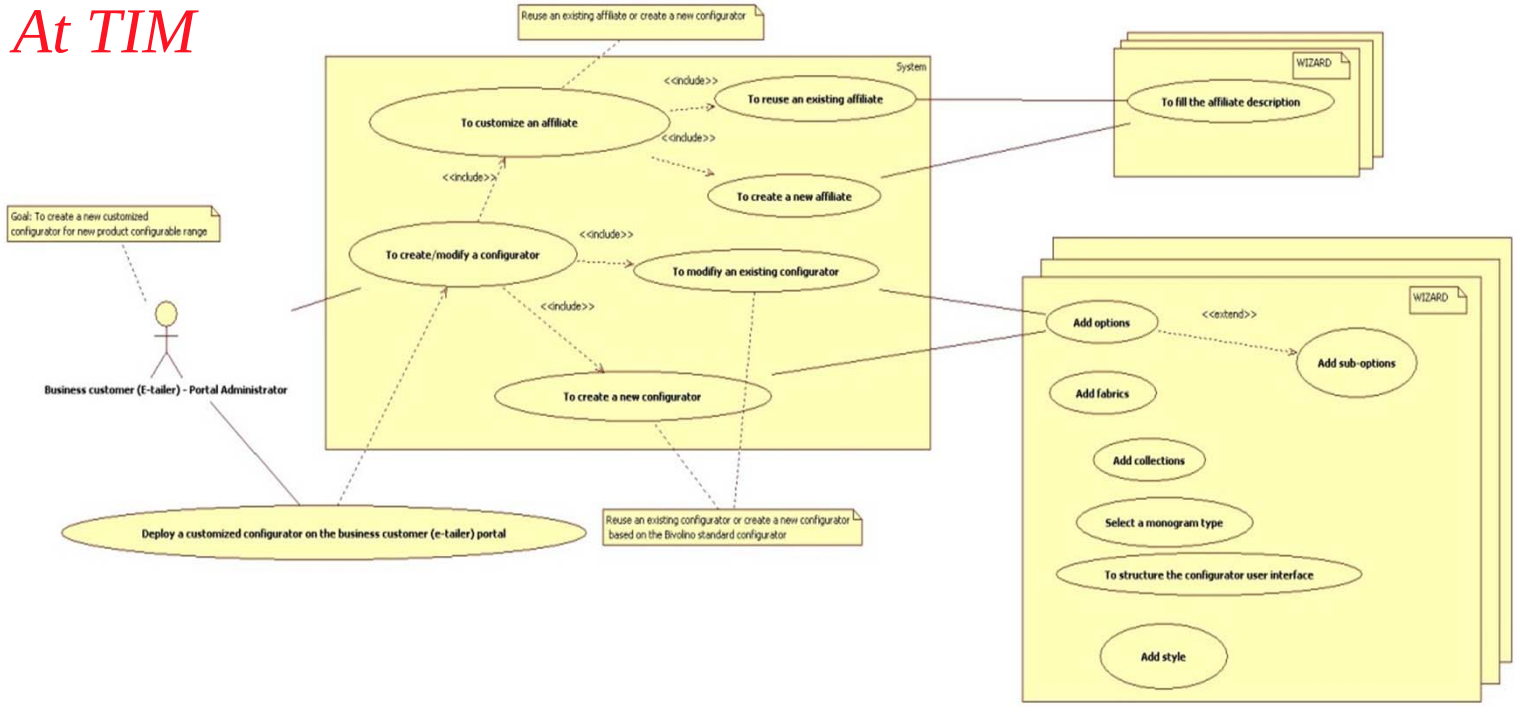
Example of BIVOLINO

At TIM *General architecture*



Example of BIVOLINO

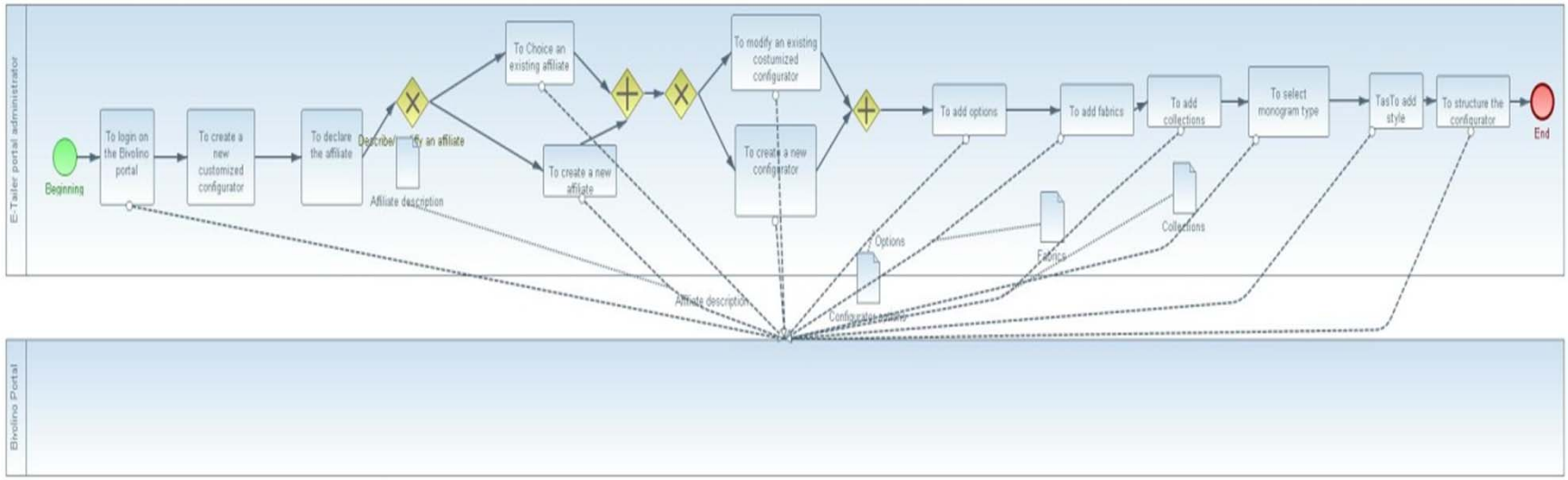
At TIM



Use case: Create a new customized configurator

Example of BIVOLINO

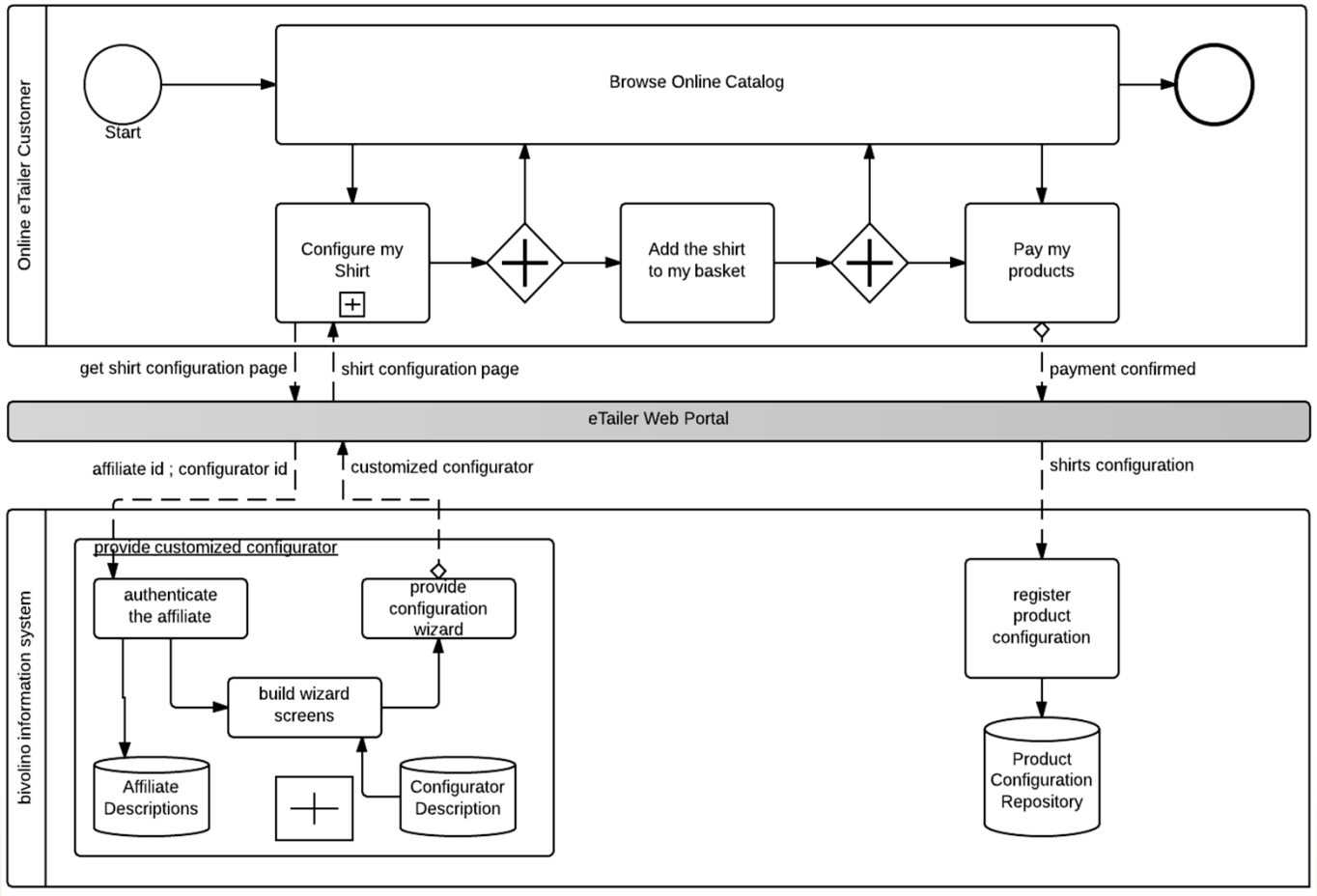
At TIM



BPMN: Create a new customized configurator (obtained by tranformation of EA* diagram)

Example of BIVOLINO

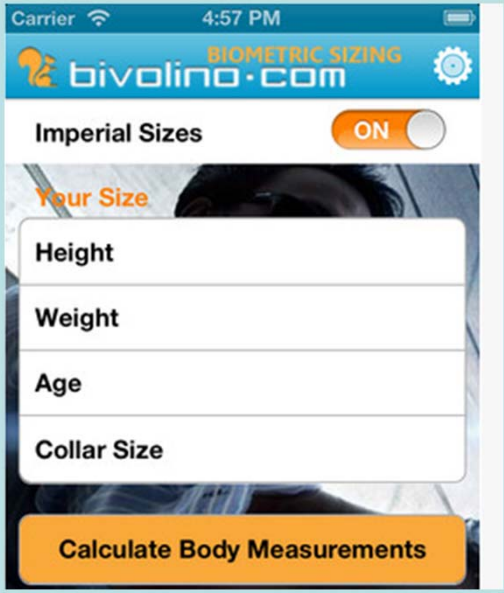
At TIM



User Process : Buy a shirt online

First results for BIVOLINO

- Mobile application enabling customers to know their shirt size providing their height, weight, collar, etc.
- The MSEE Development Platform has been used to define Technology Specific Model of the application.
- The MSEE Mobile Business Platform (Mobile Development Module) has been used to develop the application.
- **This mobile application is interoperable with the product configurator of BIVOLINO and with the IT of the shirt manufacturer**



Conclusions

- ✓ The use of enterprise modelling is crucial to understand the running of a VE and to ensure that the implementation is coherent with its strategic objectives and enables the interoperability of practices and IT
- ✓ The representation must be done at several levels of detail in a coherent architecture in order to ensure the coherence of the modelling languages
- ✓ Need to apply the MDSEA principles and related languages to validate and modify them
- ✓ Acknowledgement to